<https://github.com/test-kitchen/kitchen-vagrant>

**itchen::Vagrant**

A Test Kitchen Driver for Vagrant.

This driver works by generating a single Vagrantfile for each instance in a sandboxed directory. Since the Vagrantfile is written out on disk, Vagrant needs absolutely no knowledge of Test Kitchen. So no Vagrant plugins are required.

**Requirements**

**Vagrant**

A Vagrant version of 1.1.0 or higher is required for this driver which means that a [native package](http://www.vagrantup.com/downloads.html) must be installed on the system running Test Kitchen.

**Note:** If you have previously installed Vagrant as a gem (a version prior to 1.1.0), this version may be resolved first in yourPATH. If you receive an error message that Vagrant is too old despite having installed Vagrant as a package, you may be required to uninstall the gem version or modify your PATH environment. If you require the vagrant gem for older projects you should consider the [vagrant-wrapper](https://github.com/org-binbab/gem-vagrant-wrapper) gem which helps manage both styles of Vagrant installations ([background details](https://github.com/org-binbab/gem-vagrant-wrapper#background---aka-the-vagrant-gem-enigma)).

If you are creating Windows VMs over a WinRM Transport, then the [vagrant-winrm](https://github.com/criteo/vagrant-winrm) Vagrant plugin must be installed. As a consequence, the minimum version of Vagrant required is 1.6 or higher.

**Virtualbox, VMware Fusion/Workstation, and/or Parallels**

Currently this driver supports VirtualBox and VMware Fusion/Workstation. Virtualbox is free and is the default provider for Vagrant.

[VirtualBox package](https://www.virtualbox.org/wiki/Downloads)

If you would like to use VMware Fusion/Workstation you must purchase the software from VMware and then must also purchase the Vagrant VMware plugin.

[Vagrant VMware Plugin](http://www.vagrantup.com/vmware)

[VMware Fusion](http://www.vmware.com/products/fusion/overview.html)

[VMware Workstation](http://www.vmware.com/products/workstation/)

If you would like to use Parallels Desktop you must also purchase the software but the vagrant-parallels plugin is freely available.

[Parallels Desktop for Mac](http://www.parallels.com/products/desktop/download/)

[Vagrant Parallels Provider](https://github.com/Parallels/vagrant-parallels)

**Installation and Setup**

Please read the [Driver usage](http://kitchen.ci/docs/getting-started/adding-platform) page for more details.

**Default Configuration**

For a select number of platforms and a select number of hypervisors (VirtualBox, VMware, and Parallels) default boxes are published under the [Bento organization](https://atlas.hashicorp.com/bento) on [Atlas](https://atlas.hashicorp.com/) such as:

---

platforms:

- name: ubuntu-12.04

- name: ubuntu-14.04

- name: centos-5.11

- name: centos-6.7

- name: centos-7.2

- name: debian-7.9

- name: debian-8.3

- name: fedora-22

- name: fedora-23

- name: freebsd-9.3

- name: freebsd-10.2

This will effectively generate a configuration similar to:

---

platforms:

- name: ubuntu-12.04

driver:

box: bento/ubuntu-12.04

- name: ubuntu-14.04

driver:

box: bento/ubuntu-14.04

- name: centos-5.11

driver:

box: bento/centos-5.11

- name: centos-6.7

driver:

box: bento/centos-6.7

- name: centos-7.2

driver:

box: bento/centos-7.2

# ...

Any other platform names will set a more reasonable default for box and leave box\_url unset. For example:

---

platforms:

- name: slackware-14.1

- name: openbsd-5.6

- name: windows-2012r2

This will effectively generate a configuration similar to:

---

platforms:

- name: slackware-14.1

driver:

box: slackware-14.1

- name: openbsd-5.6

driver:

box: openbsd-5.6

- name: windows-2012r2

driver:

box: windows-2012r2

Many host wide defaults for Vagrant can be set using $HOME/.vagrant.d/Vagrantfile. See the [Vagrantfile documentation](http://docs.vagrantup.com/v2/vagrantfile/index.html) for more information.

**Configuration**

**box**

**Required** This determines which Vagrant box will be used. For more details, please read the Vagrant [machine settings](http://docs.vagrantup.com/v2/vagrantfile/machine_settings.html) page.

The default will be computed from the platform name of the instance. However, for a small number of common/known platforms in the [Bento](https://github.com/chef/bento) project, the default will prepend "opscode-" to the start to match the downloadable box\_url (see below).

For example, a platform with a Bento box called "ubuntu-14.04" will produce a default box value of "opscode-ubuntu-14.04". Alternatively, a box called "slackware-14.1" will produce a default box value of `"slackware-14.1".

**box\_check\_update**

Whether to check for box updates (enabled by default).

**box\_url**

A default URL will be computed only for a small number of common/known platforms in the [Bento](https://github.com/chef/bento) project. Additionally, a URL will only be computed if the Vagrant provider is VirtualBox or is VMware based (these are the only providers with downloadable base boxes).

**box\_version**

The [version](https://docs.vagrantup.com/v2/boxes/versioning.html) of the configured box.

The default is nil, indicating unset.

**communicator**

**Note:** It should largely be the responsibility of the underlying Vagrant base box to properly set the config.vm.communicatorvalue. For example, if the base box is a Windows operating system and does not have an SSH service installed and enabled, then Vagrant will be unable to even boot it (using vagrant up), without a custom Vagrantfile. If you are authoring a base box, please take care to set your value for communicator to give your users the best possible out-of-the-box experience.

For overriding the default communicator setting of the base box.

For example:

---

driver:

communicator: ssh

will generate a Vagrantfile configuration similar to:

config.vm.communicator = "ssh"

The default is nil assuming ssh will be used.

**customize**

A **Hash** of customizations to a Vagrant virtual machine. Each key/value pair will be passed to your providers customization block. For example, with the default virtualbox provider:

---

driver:

customize:

memory: 1024

cpuexecutioncap: 50

will generate a Vagrantfile configuration similar to:

Vagrant.configure("2") do |config|

# ...

config.vm.provider :virtualbox do |virtualbox|

virtualbox.customize ["modifyvm", :id, "--memory", "1024"]

virtualbox.customize ["modifyvm", :id, "--cpuexecutioncap", "50"]

end

end

Please read the "Customizations" sections for [VirtualBox](http://docs.vagrantup.com/v2/virtualbox/configuration.html) and [VMware](http://docs.vagrantup.com/v2/vmware/configuration.html) for more details.

**guest**

**Note:** It should largely be the responsibility of the underlying Vagrant base box to properly set the config.vm.guest value. For example, if the base box is a Windows operating system, then Vagrant will be unable to even boot it (using vagrant up), without a custom Vagrantfile. If you are authoring a base box, please take care to set your value for communicator to give your users the best possible out-of-the-box experience.

For overriding the default guest setting of the base box.

The default is unset, or nil.

**gui**

Allows GUI mode for each defined platform. Default is **nil**. Value is passed to the config.vm.provider but only for the VirtualBox and VMware-based providers.

---

platforms:

- name: ubuntu-14.04

driver:

gui: true

will generate a Vagrantfile configuration similar to:

Vagrant.configure("2") do |config|

# ...

c.vm.provider :virtualbox do |p|

p.gui = true

end

end

For more info about GUI vs. Headless mode please see [vagrant configuration docs](http://docs.vagrantup.com/v2/virtualbox/configuration.html)

**linked\_clone**

Allows to use linked clones to import boxes for VirtualBox, VMware and Parallels Desktop. Default is **nil**.

---

platforms:

- name: ubuntu-14.04

driver:

linked\_clone: true

will generate a Vagrantfile configuration similar to:

Vagrant.configure("2") do |config|

# ...

c.vm.provider :virtualbox do |p|

p.linked\_clone = true

end

end

**network**

An **Array** of network customizations for the virtual machine. Each Array element is itself an Array of arguments to be passed to the config.vm.network method. For example:

---

driver:

network:

- ["forwarded\_port", {guest: 80, host: 8080}]

- ["private\_network", {ip: "192.168.33.33"}]

will generate a Vagrantfile configuration similar to:

Vagrant.configure("2") do |config|

# ...

config.vm.network :forwarded\_port, guest: 80, host: 8080

config.vm.network :private\_network, ip: "192.168.33.33"

end

Please read the Vagrant [networking basic usage](http://docs.vagrantup.com/v2/networking/basic_usage.html) page for more details.

The default is an empty Array, [].

**pre\_create\_command**

An optional hook to run a command immediately prior to the vagrant up --no-provisioner command being executed.

There is an optional token, {{vagrant\_root}} that can be used in the pre\_create\_command string which will be expanded by the driver to be the full path to the sandboxed Vagrant root directory containing the Vagrantfile. This command will be executed from the directory containing the .kitchen.yml file, or the kitchen\_root.

For example, if your project requires [Bindler](https://github.com/fgrehm/bindler), this command could be:

---

driver

pre\_create\_command: cp .vagrant\_plugins.json {{vagrant\_root}}/ && vagrant plugin bundle

The default is unset, or nil.

**provider**

This determines which Vagrant provider to use. The value should match the provider name in Vagrant. For example, to use VMware Fusion the provider should be vmware\_fusion. Please see the docs on [providers](http://docs.vagrantup.com/v2/providers/index.html) for further details.

By default the value is unset, or nil. In this case the driver will use the Vagrant [default provider](http://docs.vagrantup.com/v2/providers/default.html) which at this current time isvirtualbox unless set by VAGRANT\_DEFAULT\_PROVIDER environment variable.

**provision**

Set to true if you want to do the provision of vagrant in create. Useful in case of you want to customize the OS in provision phase of vagrant

**ssh\_key**

This is the path to the private key file used for SSH authentication if you would like to use your own private ssh key instead of the default vagrant insecure private key.

If this value is a relative path, then it will be expanded relative to the location of the main Vagrantfile. If this value is nil, then the default insecure private key that ships with Vagrant will be used.

The default value is unset, or nil.

**synced\_folders**

Allow the user to specify a collection of synced folders on each Vagrant instance. Source paths can be relative to the kitchen root.

The default is an empty Array, or []. The example:

---

driver:

synced\_folders:

- ["data/%{instance\_name}", "/opt/instance\_data"]

- ["/host\_path", "/vm\_path", "create: true, type: :nfs"]

will generate a Vagrantfile configuration similar to:

Vagrant.configure("2") do |config|

# ...

c.vm.synced\_folder "/Users/mray/cookbooks/pxe\_dust/data/default-ubuntu-1204", "/opt/instance\_data"

c.vm.synced\_folder "/host\_path", "/vm\_path", create: true, type: :nfs

end

**vagrantfile\_erb**

An alternate Vagrantfile ERB template that will be rendered for use by this driver. The binding context for the ERB processing is that of the Driver object, which means that methods like config[:kitchen\_root], instance.name, andinstance.provisioner[:run\_list] can be used to compose a custom Vagrantfile if necessary.

**Warning:** Be cautious when going down this road as your setup may cease to be portable or applicable to other Test Kitchen Drivers such as Ec2 or Docker. Using the alternative Vagrantfile template strategy may be a dangerous road--be aware.

The default is to use a template which ships with this gem.

**vagrantfiles**

An array of paths to other Vagrantfiles to be merged with the default one. The paths can be absolute or relative to the .kitchen.yml file.

**Note:** the Vagrantfiles must have a .rb extension to satisfy Ruby's Kernel#require.

---

driver:

vagrantfiles:

- VagrantfileA.rb

- /tmp/VagrantfileB.rb

**vm\_hostname**

Sets the internal hostname for the instance. This is not used when connecting to the Vagrant virtual machine.

For more details on this setting please read the [config.vm.hostname](http://docs.vagrantup.com/v2/vagrantfile/machine_settings.html) section of the Vagrant documentation.

To prevent this value from being rendered in the default Vagrantfile, you can set this value to false.

The default will be computed from the name of the instance. For example, the instance was called "default-fuzz-9" will produce a default vm\_hostname value of "default-fuzz-9". For Windows-based platforms, a default of nil is used to save on boot time and potential rebooting.

**Development**

* Source hosted at [GitHub](https://github.com/opscode/kitchen-vagrant)
* Report issues/questions/feature requests on [GitHub Issues](https://github.com/opscode/kitchen-vagrant/issues)

Pull requests are very welcome! Make sure your patches are well tested. Ideally create a topic branch for every separate change you make. For example:

1. Fork the repo
2. Create your feature branch (git checkout -b my-new-feature)
3. Commit your changes (git commit -am 'Added some feature')
4. Push to the branch (git push origin my-new-feature)
5. Create new Pull Request